Subject: Re: passing parameters from base to base

Posted by David Fanning on Thu, 28 Nov 2002 17:38:29 GMT

View Forum Message <> Reply to Message

JD Smith (jdsmith@as.arizona.edu) writes:

> You put your finger right on the tricky bit (good eye ;). Right now,
> my solution is to "introduce" objects which want to communicate to
> each other on a higher level. E.g. I might say in my wrapper program:
>
> oDraw=obj_new('tvDraw')
> oFunk=obj_new('tvFunky',oDraw)
> oNother=obj_new('tvNother',oDraw)
>
> and oFunk could sign up for the tvDraw messages it needs. Perhaps
> oFunk even builds an ObjMsg object of its own and signs up with it
> directly. Usually this is perfectly adequate: relations and
> introductions can be done at compile time or Init time. This works
> well for, e.g., building a viewer shell with all the desired plug-ins,
> especially when you need to carry widget bases around to build the
> interface inside.

The object-widget system Dave Burridge and I have developed
is similar. We use the notion of an object hierarchy, almost
identical to a widget hierarchy. Our objects can find each
other to pass messages back and forth by searching through
the hierarchy, recursively if necessary. It is similar to
the "find by name" mechanism in widgets.

When we spawn other top-level bases from our application,
they can be given an "event object", which is simply a list
of objects who would like to know about what they are up do.
Events can be diverted to other objects, or the other objects
can simply be notified that an event has occurred. For example,
our draw widget object has been written in such a way that you
can "plug in" event handler objects that interpret events from
the draw widget in many different ways, allowing us to put the
draw widget in "zoom mode" or "report mode" or "draw mode", etc.

As we work with this system, we find stronger and stronger
parallels with the widget system. In fact, we were wondering
the other day if we shouldn't give our objects unique IDs, just
like widgets, so we can keep a list somewhere (an XManager object,
say) that will allow us to find them even more easily, wherever
they happen to be.

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155